Research Article

Internet of Things Network Topology Discovery Algorithm Based on Wireless Sensors

Hao Sun

College of Engineering and Technology, Xi’an FanYi University, Xi’an, 710105 Shaanxi, China

Correspondence should be addressed to Hao Sun; sharp_syc@126.com

Received 13 January 2022; Revised 23 February 2022; Accepted 8 March 2022; Published 8 April 2022

Academic Editor: Mohamed Elhoseny

Copyright © 2022 Hao Sun. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Remote sensor network is a multijump self-coordinating organization framework shaped by countless energy-restricted miniature sensor hubs sent in the observing region. Be that as it may, the correspondence conventions of customary organization configuration cannot be straightforwardly applied to remote sensor organizations. The improvement of committed correspondence conventions and estimation techniques has turned into a pressing exploration subject in the field of remote sensor organizations. The research purpose of this paper is to research the Internet of Things network topology discovery algorithm based on wireless sensors. This paper analyzes and mines an adaptive neighbor discovery scheduling algorithm with lower latency, acquires the information of potential neighbor nodes based on existing neighbor nodes, discovers potential neighbor nodes by proactively waking up, and studies the information recommendation mechanism between neighbor nodes and compares. The intimacy between neighbor nodes (such as common neighbor rate) is used to selectively receive recommended information from neighbor nodes, thereby filtering redundant data information, lessening hub energy utilization, and accomplishing the motivation behind broadening the organization life cycle. A geography disclosure calculation in light of versatile specialists is proposed to tackle the geography revelation issue under the various leveled geography structure in remote sensor organizations. The calculation joins the qualities of portable specialist innovation and various leveled Internet of Things network geography and simultaneously further develops the relocation methodology of versatile specialists for information assortment, intermittent undertaking assignment, and organization status observing. Research shows that in the investigation of the effect of hub obligation cycle on framework execution, the energy utilization of the DDC-Group calculation is marginally expanded contrasted and the gathering calculation, and the increase is not more than 1%, but compared to the CNR-Group increase more.

1. Introduction

Remote sensor network incorporates current sensor innovation, microelectronic innovation, correspondence innovation, installed figuring innovation, and appropriated data handling innovation and different disciplines. It is an arising crossresearch field and is applied to military surveillance, natural observing, clinical wellbeing and space investigation, and so forth, overrun all degrees of life, particularly in the fields of public protection and serious debacle occasion checking, which will assume a vital part in the modernization of our country. Topological sort is to sort the nodes on a DAG (Directed Acyclic Graph). Simply put, it is to pull the DAG into a chain without destroying the order of nodes. If you take the technology tree in the game as an example, topological sorting is to find a possible order of the technology tree.

Numerous colleges and research establishments abroad have led a ton of exploration on the examination on the geography revelation calculation of the Internet of Things network in view of remote sensors. For example, Shahzad et al. proposed a topology discovery algorithm based on a mesh network. The method is divided into two steps: one is the diffusion phase, which coordinates the nodes to broadcast the topology request message. After each node receives and rebroadcasts this message, it constructs local neighbor information. Update the corresponding data structure; the second is the collection phase, and all nodes forward the
local neighbor information to the coordinating node [1]. Nesa and Banerjee proposed a distributed topology discovery strategy based on mobile agents. This method cannot provide instantaneous Internet of Things network topology. It takes a long time to find a complete Internet of Things network topology and needs to distribute a lot of messages; so, it is not efficient and consumes bandwidth [2]. Dhanvijay and Patil proposed the ad hoc network management protocol ANMP, which uses a distributed set of nodes to maintain the information of the node and its neighbors, attempts to integrate the characteristics of the SNMP protocol, and uses a layered mechanism to collect topology information. The cluster head node is dynamically selected due to geographic location or connection conditions, and topology information is collected through the MIB in the cluster head [3].

In our country, Wei and Zhou utilizes the hub development model in the sensor organization to foresee the potential neighbor hubs inside the correspondence scope of the hub and progressively changes the obligation pattern of the hub as per the quantity of potential neighbor hubs, so as to find neighbor nodes faster and more. Aiming at the problem of network coverage optimization of the sensor node model with fixed sensing angle and sensing radius used in the past, he proposed a new node model with variable sensing angle and sensing radius [4]. Fu et al. proposed an adaptive RSSI discovery scheme, which improves the accuracy of discovery in the ZigBee network by studying the communication channel state between two nodes, and uses a Markov model to correct the RSSI scale caused by factors such as noise error [5]. Khousi et al. proposed a distributed discovery algorithm based on MDS-MAP. This algorithm distributes computing tasks to different nodes for distributed computing by defining node classes, reducing the amount of computing tasks and reducing the energy consumption of nodes [6].

The purpose of this paper is to comprehensively, accurately, and quickly perform network topology discovery for IoT network management, fault location, and congestion control. This paper is based on the neighbor discovery algorithm of dynamic duty cycle. Based on the full analysis of Disco, U-Connect, and other algorithms, a neighbor revelation calculation in light of dynamic obligation cycle DDC-Group is proposed for the proper obligation cycle issue of existing neighbor disclosure calculations, and the genuine hub development model is utilized to anticipate potential neighbor hubs, powerfully change the obligation pattern of hubs through the quantity of potential neighbor hubs, and suitably expand the enlivening season of hubs to screen and find neighbor hubs quicker and all the more rapidly, so as to improve the efficiency of neighbor node discovery and reduce discovery delay. Finally, it is compared with Disco, Group, and CNR-Group algorithms through simulation experiments. The neighbor discovery algorithm drives the node to actively discover neighbor nodes when it wakes up, avoiding the long waiting delay of the traditional passive neighbor discovery algorithm, thereby realizing the rapid discovery of neighbor nodes and reducing the delay of neighbor discovery.

2. Internet of Things Network Topology Discovery Algorithm Based on Wireless Sensors

2.1. Overview of the Cluster Routing Algorithm for Wireless Sensor Networks and Mobile Agent Technology

2.1.1. LEACH Cluster Routing Algorithm. Drain mostly haphazardly chooses the group head hub through a roundabout technique to adjust the energy utilization of the whole organization. It characterizes the idea of “round.” Each round comprises of three phases: group head determination, bunch development, and stable information transmission [7, 8]. The LEACH (Low Energy Adaptive Clustering Hierarchy) calculation is a versatile bunching geography calculation. Its execution interaction is occasional, and each cycle is isolated into a bunch foundation stage and a steady information correspondence stage.

The group head determination stage is answerable for choosing all bunch head hubs in this round. The primary determination guideline is as follows: every sensor hub produces an irregular number somewhere in the range of 0 and 1. Assuming the irregular number is not exactly the characterized limit $T(n)$, it is chosen as the bunch head and distributes the transmission message that it turns into the group head; if not, it is a part hub.

$$T(n) \begin{cases} p & n \in G \\ 0 & \text{Other} \end{cases} \quad (1)$$

Among them, $p$ is the level of the quantity of bunch heads in all hubs in the organization, $r$ is the quantity of current determination rounds, and $G$ is the arrangement of hubs that are not group heads in the latest $1/p$ round [9, 10].

2.1.2. GAF Cluster Routing Algorithm. Geological Adaptive Fidelity (GAF) is an energy-effective directing calculation in specially appointed organizations. GAF is a method for saving energy by keeping hubs as off as could really be expected. The node obtains its own “location” in the network through GPS positioning and is thus classified into the corresponding cell. If the "location" of two nodes is the same, they are considered to be equivalent in routing; each cell selects a cluster head regularly, only the cluster head remains active, and other nodes go to sleep. The GAF algorithm is mainly divided into two stages:

1. **Divide Cells.** Divide the network into several adjacent virtual cells. The maximum distance $R$ between any two points in every two adjacent cells ($R$ is the wireless signal communication radius) and the cell side length $r$ satisfy the following formula condition:

$$r^2 + (2r^2) \leq R^2 \rightarrow r \leq \frac{R}{\sqrt{3}}. \quad (2)$$

Condition (2) guarantees that the distance between all hubs between two neighboring cells is not exactly their
2.2. Wireless Sensor Network Routing Algorithm Based on Effective Clustering NDEA

2.2.1. Network Model and Related Definitions. To work with research, we make the accompanying suspicions:

(1) A remote sensor network contains a sink hub and a few sensor hubs. These hubs are haphazardly appropriated in a specific region. It is expected that the sensor hubs and sink hubs are static, and the sensor hubs have consistently sent information to the sink hubs.

(2) The channel between sensor hubs is a symmetric double channel, or at least, hub A can speak with hub B, and hub B can likewise speak with hub A. Because of the distinction in the sending power and getting responsiveness of the sensor hubs, the existence of a unidirectional connection is not thought of, and the correspondence between the hubs is single-bounce.

(3) Definition of energy model.

(4) The issue of energy misfortune is the way in to the plan of the bunch directing calculation in remote sensor organizations. The NDEA calculation utilizes a similar energy model as the LEACH calculation. The energy utilization of a sensor hub can be communicated as

\[
E_i = E_{Tx} + E_{Rx}
\]

\[
E_{Tx} = E_{elec} \times I + \epsilon_{amp} \times I \times d^3,
\]

\[E_{Rx} = E_{elec} \times I,
\]

where \(E_i\) is the absolute energy consumed by the sensor hub, \(E_{Tx}\) and \(E_{Rx}\) are the energy consumed by sending and getting 1-digit information, \(E_{elec}\) is the energy consumed by the working of the communicating circuit, \(\epsilon_{amp}(J/\text{bits/m}^2)\) is the power amplifier energy consumed, \(I\) is the length of transmitted information bits, \(\lambda\) is the path loss, and \(d\) is the correspondence distance between the beneficiary and the transmitter. As indicated by the distance between the shipper and the beneficiary, the power enhancement misfortune takes on the free space model and the multipath blurring model to decide the worth of \(\lambda\). While the sending distance is short, the free space model is applied, and \(\lambda\) is 2, while the sending distance is longer. When the multipath blurring model is utilized, \(\lambda\) is 4, and the energy model is displayed in Figure 1 [13, 14].

\[
\Psi(i,j) = \Psi(j,i) = 1 (\forall i, j|1 \leq i, j \leq n, \exists t \leq L).
\]

2.2.2. The Basic Idea of NDEA. In the NDEA calculation, a weight work \(W(n)\) is characterized for every hub. The meaning of weight includes two factors, the neighbor hub degree and the excess energy of the hub. The weight work is communicated as

\[
W(n) = w_1f(d_n) + w_2g(\epsilon_n)
\]

where \(w_1\) and \(w_2\) are weighting parameters, the weighting parameters can be adjusted according to different applications, \(f(d_n)\) is a function of node \(n\)'s neighbor node degree, and \(g(\epsilon_n)\) is an element of hub \(n\) in view of leftover energy, the hub with the littlest weight. It is considered as a competitor hub that can turn into the group head [5, 15].

2.3. Disco Algorithm

2.3.1. Principle of the Disco Algorithm. The process of neighbor discovery is as follows: a node in the network is within the communication range of another node, and at a certain moment, the two nodes are in the awake state at the same time. If one node is broadcasting information by radio, the other node can be awake. After receiving the broadcast information, the two nodes exchange data information, complete neighbor discovery, and become each other’s neighbor nodes. The calculation begins from the hub with the biggest degree in the organization, joins neighbor hub search and neighbor hub casting a ballot, grows the hunt from the nearby piece of the organization to the entire, and lastly shapes numerous disjoint networks. The running season of the calculation is near \(O(m + n)\).

2.3.2. Network Model Design

(1) Related Definitions. Accepting that there are \(n\) sensor hubs in a remote sensor organization, the neighbor disclosure issue includes the development of \(\Psi(m, t)\) (where \(1 \leq m \leq n\)). The development of \(\Psi(m, t)\) should guarantee that any \(A\) couple of hubs \(i\) and \(j\) (where \(i \neq j\)); at whatever point, they are inside the correspondence scope of one another, and the time they meet surpasses \(L\) unit time; they can see one another. Among them, \(L\) represents the delay requirement of neighbor discovery [16, 17]. The neighbor discovery problem is expressed as the following relationship using discovery scheduling \(\Psi(m, t)\):

\[
\Psi(i, t) = \Psi(j, t) = 1 (\forall i, j|1 \leq i, j \leq n, \exists t \leq L).
\]

The simultaneous symmetric neighbor disclosure issue with a postpone prerequisite of \(L\) is characterized as a similar revelation timetable and introduction time allocated to \(n\)
sensor hubs in the organization, and the accompanying relationship is fulfilled:

\[ \Psi(i, t) = \Psi(j, t) \quad (\forall t \forall 1 \leq i, j \leq n). \quad (6) \]

Moreover, the maximum discovery delay in the neighbor discovery process cannot exceed \( L \). The offset symmetric neighbor disclosure issue with a defer necessity of \( L \) is characterized as a similar revelation timetable and relative stage offset given to \( n \) sensor hubs in the organization, and the accompanying relationship is fulfilled:

\[ \Psi(i, t) = \Psi(j, \phi_{ij}) \quad (\forall t \forall 1 \leq i, j \leq n). \quad (7) \]

Moreover, the maximum discovery delay during the neighbor discovery process cannot exceed \( L \). In formula (7), \( \phi_{ij} \) represents the relative displacement offset of node \( i \) and node \( j \) at the original time \( t \) [18, 19].

(2) Metrics. In order to better evaluate the quality of a neighbor discovery algorithm and whether it has certain practical significance, we will use the energy consumption-delay product \( M \) as the evaluation criterion of the neighbor discovery algorithm [20, 21]. To get done with different responsibilities, sensor hubs need to finish the assortment and change of checking information, information the board and handling, reaction to the errand solicitation of the sink hub, and hub control and different assignments. Moving hubs can prompt incoherent work content. For the neighbor discovery scheduling \( \Psi(m, t) \) with a period of \( T \), the following relationship is satisfied:

\[ \Psi(m, t) = \Psi(m, t + T)(\forall). \quad (8) \]
Then, the average energy consumption $P$ in a scheduling period is expressed as

$$P = \frac{1}{T} \int_0^T \Psi(m, t) dt. \quad (9)$$

In this neighbor disclosure booking $\Psi(m, t)$ with a time of $T$, expecting that the most obviously awful revelation delay among all sets of (hubs give all conceivable relative stage balances) is $L$, then we will energy-delay. The product $M$ is defined as

$$M = PL = \frac{L}{T} \int_0^T \xi(m, t) dt. \quad (10)$$

Due to the slot nature of the actual neighbor discovery algorithm, the discovery scheduling is discrete; so, here, we use $M$ to represent the discovery scheduling in the discrete state and the node $m$ with a period of $T$ that meets the delay requirement, $L$ is in the time slot $t$, the discovery schedule can be expressed as $\xi(m, t)$, and the average energy consumption $P$ in a period is

$$P = \frac{1}{T} \sum_{t=0}^{T-1} \xi(m, t) dt. \quad (11)$$

Then, the energy consumption-delay product $M$ is

$$M = PL = \frac{L}{T} \sum_{t=0}^{T-1} \xi(m, t) dt. \quad (12)$$

2.4. Neighbor Discovery Algorithm Based on Dynamic Duty Cycle. On account of a decent obligation cycle, the wake-up time and rest season of the hub are fixed. At the point when the obligation pattern of a hub is in an evolving state, the wake-up time and rest time will change appropriately. The neighbor revelation calculation in light of dynamic obligation cycle is an exhaustive neighbor hub disclosure calculation, which depends on the low-dormancy neighbor disclosure calculation. During the revelation interaction of neighbor hubs, the obligation pattern of the sensor hub is not generally fixed. The hubs in the remote sensor network are occasionally in the wake-up state, and the hubs in this functioning mode can exist together with the hubs in other working modes and help the hubs in other working modes to work.

Whenever a sensor node moves from old position to new position, the duty cycle needs to be adjusted in real time to monitor and discover potential neighbor nodes. Assuming that the communication radius of the sensor node is $R$, the node’s moving speed is $v$, and the node $S$ reaches $S’$ after time $t$, the distance the node moves:

$$\text{Distance}_{S \rightarrow S'} = v \times t. \quad (13)$$
The area where the communication circles intersect is $\Delta S$, and the area of the newly added area is $\Delta S'$:

\[
\Delta S = 2\left(\frac{3\theta}{2} \cdot R^2 - S_{SAB}\right) = 2(\theta \cdot R^2 - S_{SAB}).
\]
\[
S_{SAB} = \frac{1}{2} \cdot 2R \sin \theta \cdot \frac{1}{2} \cdot \text{Distance}_{S \rightarrow S'} = \frac{1}{2} \cdot R \sin \theta \cdot \text{Distance}_{S \rightarrow S'},
\]
\[
\cos \theta = \frac{1}{2} \cdot \text{Distance}_{S \rightarrow S'} = \frac{\text{Distance}_{S \rightarrow S'}}{2R},
\]
\[
\theta = \arccos \frac{\text{Distance}_{S \rightarrow S'}}{2R}.
\]

From the above formula, the area of the new area can be obtained:

\[
\Delta S' = \pi R^2 - \Delta S.
\]

Expecting that the dispersion $P$ of sensor hubs in a versatile sensor network complies with the Poisson circulation of $\lambda$, or at least, $P \sim P(\lambda)$, when the sensor hub moves to another position, the quantity of neighbor hubs found by the CNR-Group calculation in the recently added region is $N$.

The normal worth of the quantity of neighbor hubs in the recently added correspondence region $S'$ is

\[
E\left(S'\right) = \lambda \Delta S'.
\]
Then, at that point, the quantity of neighbor hubs not found in the recently added correspondence region is

\[
E(S_{no}') = E(S') - N = \lambda \Delta S' - N. \tag{17}
\]

In recipe (17), \(\lambda\) is the boundary of Poisson dissemination, \(\lambda \Delta S'\) addresses the quantity of neighbor hubs in the recently added correspondence region, and \(N\) addresses the low-inactivity neighbor revelation calculation CNR-Group has found during hub development the quantity of neighbor hubs.

Assume it is tracked down that the wake-up season of hub DN is \(T_{active}\) and the wake-up period is \(T\). Simultaneously, the wake-up time \(T_{active}\) is viewed as a container, and the likely \(E(S')\) neighbor hubs in the recently added correspondence region are viewed as balls. As indicated by the ball and box model, it should be visible that the likelihood that the expected \(E(S_{no}')\) neighbor hubs not found in the \(S'\) area will not awaken inside the wake-up season of the DN hub that is

\[
p_{active} = 1 - p_{active\ no}\). \tag{19}
\]

It can be seen from equation (19) that the probability of a node in the area \(S'\) awakening within \(T_{active}\) time is \(p_{active}\), which means that there are still some nodes that will not wake up within \(T_{active}\) time. At this time, it is necessary to adjust the discovery node DN the duty cycle to monitor undiscovered neighbor nodes.

At this time, the wake-up time \(T'_{active}\) of the discovery node DN is extended to \(T'_{active}\):

\[
T'_{active} = T_{active} + T_{active} \times p_{active\ no}. \tag{20}
\]

Then, it is found that the duty cycle DC of the node DN is adjusted to \(DC'\):

\[
DC = \frac{T_{active} + T_{active} \times p_{active\ no}}{T} \times 100\%. \tag{21}
\]

Among them, the wake-up period \(T\) is the sum of the wake-up time and the sleep time, namely,

\[
T = T_{active} + T_{dormant}. \tag{22}
\]

From the above investigation process, it very well may be seen that the obligation pattern of the hub DN has expanded, that is to say, the time that the hub is in the conscious state increments, and the hub is found to utilize the expanded waking chance to screen and find potential neighbor hubs in the recently added region. The disclosure of the hub the deferral is diminished; however, as the obligation pattern of the hub expands, the energy utilization of the hub should increment.
3. Test Research on the Internet of the Things
Network Topology Discovery Algorithm Based on Wireless Sensors

3.1. Simulation Environment. The algorithm simulation environment is Windows XP + cygwin+NS_2.27 and the simulation network parameter setting: 60 wireless sensor nodes are distributed in an area of 100 m × 100 m and are divided into 6 clusters. The mobile agent is randomly generated on the 6 cluster head nodes, where Code_Size is set to 800 bytes; Data_Size is 1000 Byte; State_Size is 400 bytes; the time delay factor required for the serialization of unit byte data is \(2^3 \times 14 - 9\); the creation of mobile agent. The delay \(T_c\) and the processing delay \(T_p\) are ignored. The mobile agent sending delay is \(T_s + T_p = \text{sizef} \times \text{actor} = 6.25\) ms, and the receiving delay is \(T_{\text{wait}} = 5.34\) ms and \(T_{\text{wait}} = 40\) ms.

3.2. Simulation of Mobile Agent Platform. The simulation of the mobile agent platform mainly includes two parts: the mobile agent entity and the execution environment context of the mobile agent. A mobile agent is a software entity that contains code, data, and status.

3.2.1. Simulation of Mobile Agent. The simulation of the mobile agent must have the following modules: the state of the mobile agent, the running module, the event response processing module, and the environment interaction module. The life cycle of a mobile agent entity begins with creation or cloning and then begins execution and enters the active state. During the execution, the mobile agent entity may be dispatched to other nodes, converted to inactive state, or converted from inactive state to active state.

3.2.2. The Execution Environment of the Mobile Agent Context. Context is similar to the data source generator in NS2 (such as FTP and CBR) and is implemented in the application layer of NS2 network simulation. Context includes the following modules: management module, transmission communication module, response processing module, interface module, and execution module. The interface module mainly coordinates and controls other modules and provides basic services such as creation, migration, and message transfer required by the mobile agent; the execution module is responsible for activating the execution of the mobile agent. The node movement model is described as the node first randomly selects a position in the entire moving area as the initial position and randomly selects the size and direction of the movement speed and the moving distance.

3.3. Algorithm Design and Data Collection. During the discovery process of neighbor nodes, the duty cycle of sensor nodes is no longer fixed. During the movement of nodes, sensor nodes dynamically adjust their duty cycle in real time according to the status of surrounding nodes. When a node is found to move from its original position to a new position, at this time, during the time when the node is found to be awake, some of the nodes are still in a dormant state, and the node cannot be found.

In a sensor network with \(n\) sensor hubs, take hubs \(a, b,\) and \(c\) as specific illustrations, where hub \(a\) is the hub that needs to find other neighbor hubs, that is, the discovery node, and the three nodes are each scheduled according to their own work arrangements. In the process of node neighbor discovery and movement, the duty ratio of the node is
4. Test Research and Analysis of the Internet of Things Network Topology Discovery Algorithm Based on Wireless Sensors

4.1. Data Transfer Volume. To investigate the exhibition of the calculation, this article reenacts under full burden; that is to say, every hub has information shipped off the sink whenever. This article compares and analyzes the NDEA algorithm, LEACH, and Disco algorithm, and the experimental research from the aspect of data transmission is shown in Table 1.

Figure 2 shows the comparison between the improved NDEA algorithm, the original LEACH algorithm, and the sink receiving data volume of the Disco algorithm. In the simulation process, the data sent by the node to other nodes is actually its own ID number; so, the information got before the group head sends information to the sink is the rundown of IDs of the hubs that send information to itself, and the bunch head ships off the sink. This is the ID list; lastly, sink counts the quantity of information sent by every hub in light of the got information.

4.2. NDEA Algorithm and LEACH Algorithm Life Cycle Experiment Analysis. In the simulation process, the simulation data is collected regularly. The reenactment information chiefly incorporates the quantity of hubs right now making due in the organization, the aggregate sum of energy presently consumed by the organization, and how much information as of now gotten by the sink hub. This paper contrasts and breaks down the NDEA calculation and the LEACH calculation and the Disco calculation, and behaviors test research from the organization life cycle. The outcomes are displayed in Table 2.

Figure 3 shows the comparison of the number of surviving nodes in the improved NDEA algorithm and the original LEACH algorithm over time. Since NDEA needs to collect network information during network initialization, which consumes more energy than LEACH, the death of the first node occurs in NDEA at 45 s, before LEACH. Then, the overall network life cycle is about 1.6 times that of LEACH, which is better than LEACH. The curve comparison in the figure shows that NDEA can save energy consumption and make energy loss more evenly distributed to all nodes, extending the life cycle of the network.

4.3. Data Transmission Volume. Table 3 shows the comparison of the sink receiving data volume between the improved NDEA algorithm and the original LEACH algorithm.

As displayed in Figure 4, in the reenactment interaction, the information sent by a hub to different hubs is really its own ID number; so, the information got before the bunch head sends information to the sink is the ID of the hub that sent information to itself list, and the group head sends this ID rundown to the sink; lastly, sink counts the quantity of information sent by every hub in view of the got information. This article shows the examination of the factual data of the information sent by the hubs under the two calculations. Since the organization lifetime of the NDEA calculation is fundamentally longer than that of the LEACH calculation, the information transmission volume should likewise be higher than that of LEACH.

4.4. Influence of Node Density on System Performance. This gathering of examinations concentrates on the effect of the thickness of sensor hubs in remote sensor networks on framework execution. The presentation boundaries of the framework incorporate the disclosure postponement and energy utilization of the sensor organization. The trial results are displayed in Table 4.

The outcomes in Figure 5 show that when the thickness of hubs expands, the energy utilization of the four calculations keeps on expanding. According to the pattern perspective, the energy utilization of the DDC-Group calculation expands the most.

4.5. Effect of Node Duty Cycle on System Performance. The following investigation in this paper is to concentrate on the impact of the hub’s obligation cycle on framework execution, including the revelation of postponement and energy utilization. The test results are displayed in Table 5.
The outcomes in Figure 6 show that when the obligation pattern of the hub builds, any of the four calculations, the framework’s disclosure defer diminishes; yet, the level of effect is unique. It very well may be seen from the figure that the revelation postponement of the Disco calculation is altogether more noteworthy than the group, CNR-Group, and DDC-Group calculations under various obligation cycles, and the DDC-Group calculation is superior to the group and CNR-Group calculations.

The following examination is the connection between energy utilization and hub broadcast abnormality. The exploratory outcomes are displayed in Table 6.

Figure 7 shows that as the obligation pattern of the hub builds, the energy utilization of the hubs under the four calculations increments, which is close to a linear relationship. However, the energy consumption of the Group, CNR-Group, and DDC-Group algorithms changes more than the Disco algorithm increases more; compared with the group algorithm, the energy consumption of the DDC-Group algorithm is slightly increased, and the increase is not more than 2%, but it increases more than the CNR-Group.

4.6. Impact of Node Broadcast Irregularity on System Performance. This experiment studies the impact of node broadcast irregularity on system performance. In this group of experiments, the irregularity of node broadcasting varies from 0 to 85%. The experimental results are shown in Table 7.

It very well may be seen from Figure 8 that as the abnormality of hubs builds, the disclosure postponement of the four calculations increments to changing degrees. The Group, CNR-Group, and DDC-Group calculations are altogether better compared to the Disco calculation, and the DDC-Group calculation is the most excellent; the postponement is viewed as the least; as far as energy utilization, the DDC-Group calculation consumes the most energy.

5. Conclusions

This paper adopts the demand-wakeup mechanism of active awakening. According to the information of potential neighbor nodes obtained by existing neighbor nodes, potential neighbor nodes are discovered through active awakening, and on this basis, the recommendation mechanism of neighbor node information is studied. High-rate neighbor recommendation mechanism. Through this recommendation mechanism, the CNR-Group algorithm reduces latency and energy consumption, improves network performance, and extends network life cycle.

This paper proposes a geography disclosure calculation in light of versatile specialist. This calculation joins the attributes of versatile specialist innovation and grouping geography. The CNR-Group calculation actually beats the issues of the geography calculation, like enormous link length and establishment responsibility, significant weight on the focal hub, and low appropriated handling limit of each site. In each group, the bunch head is liable for gathering important data of the hubs in the group, including the present status and remaining energy. The mobile agent between clusters is responsible for discovering the cluster heads. Link information improves the migration strategy of mobile agents. Compared with other topology discovery algorithms based on mobile agents, this algorithm has improved globality and convergence speed, effectively reducing redundant data transmission, reducing communication consumption.

In this paper, the genuine hub development model is utilized to anticipate potential neighbor hubs, and the obligation pattern of the hub is powerfully changed through the quantity of potential neighbor hubs, to work on the productivity of neighbor hub revelation and diminish the disclosure delay. Actively waking up according to the exchanged node scheduling information can realize the discovery of neighbor nodes.

Data Availability

The data underlying the results presented in the study are available within the manuscript.

Conflicts of Interest

The authors declared no potential competing interests in our paper.

Authors’ Contributions

All authors have seen the manuscript and approved to submit to your journal.

Acknowledgments

This work was supported by the school-level research team: “Artificial Intelligence Technology Application Research” in Xi’an FanYi University in 2021. Project Number: XFU21KYTD02.

References


